2.12 CLUTTER

Numerous code discrepancies were found in the Clutter Functional Element (FE), ranging from minor to major impacts on model accuracy and operation. Some of these errors relate to clutter reflectivity estimation, geometry computation and antenna response to clutter signal. An overflow condition was found to exist due to one of the geometry errors. In addition, portions of system specific data were found to be either missing or incorrect in the data files.

There is some unnecessary code and unused variables, but, overall code quality is adequate. In addition, internal documentation is sufficient but could be improved in several specific incidents.

The table listed below summarizes the desk-checking and software testing verification activities for each subroutine in the Clutter FE. The two results columns contain checks if no discrepancies were found. Where discrepancies were found, the desk check results column contains references to discrepancies listed in Table 2.12-4, while the test case results column lists the number of the relevant test case in Table 2.12-6. More detailed information on the results is recorded in these tables.

TABLE 2.12-1. Verification Results Summary.

DESIGN ELEMENT	CODE LOCATION	DESK CHECK RESULT	TEST CASE ID	TEST CASE RESULT
12-1 Clutter Signal in Sum and Difference Channels	INICLT 140-143 SUMCLT 167-185 213-220 240-245	D1	12-3, 41, 42, 43	12-42
12-2 Range Ambiguity	INICLT 136-137 DETGEO 111,115,150 RANMIN 87-108	D2 D3 D4	12-3, 13-21	12-3, 16, 17, 20, 21
12-3 Clutter Patch Area	DETGEO 111-153 ANGWDT 42-73 SUMCLT 162,167,238	÷	12-13, 14,21, 24-28, 39,40, 42	÷
12-4 Clutter Reflectivity - Native Mode	CLUTIN 55-61 SIGMA0 87, 90-95	D5	12-6,7, 29,30	12-6, 7

TABLE 2.12-1. Verification Results Summary. (Contd.)

DESIGN ELEMENT	CODE LOCATION	DESK CHECK RESULT	TEST CASE ID	TEST CASE RESULT
12-5 Land Clutter Coefficient - Native Mode	SIGMA0 98-107 111-126 GETTTD 11-13	D6 D7	12- 31-35, 38	12-31
12-6 Terrain Shadowing - Native Mode	SIGMA0 130-135	÷	12-36	÷
12-7 Clutter Power Summation and Conversion	CLUTTR 191-193 MGTOVO 97-109	÷	12-11, 12,23	÷
12-8 Coordinate Conversion for GRACE	SUMCLT 196	÷	12-44	÷
Initialization of Clutter Variables	INICLT 112-133, 150-156, 167-172 CLUTTR 143-147	÷	12-1,2, 4,5,8,38	12-4
Clutter Management	CLUTTR 150, 152-189 MGTOVO 84-94	÷	12-9, 10-12, 22,37	12-22, 37

2.12.1 Overview

Clutter may be defined as the signal return from unintended targets. The most significant clutter signals are due to reflections from the earth surface. The intensity of the clutter echo is calculated using the standard radar equation, with target cross section replaced by the effective radar cross section of the illuminated clutter object. The effective radar cross section of the clutter patch (σ_o) is the product of the ground area being illuminated by the radar and the reflection coefficient of the earth surface. The reflection coefficient is typically an empirically derived value dependent upon the surface characteristics. The illuminated area is typically represented as a segment of a circular area, bounded by the azimuthal antenna beam width and within the range illuminated during a single pulse.

ESAMS implementation of the Clutter Signal FE is done primarily in Subroutines CLUTIN, INICLT, GETTTD, ANGWDT, DETGEO, RANMIN, SIGMA0, SUMCLT, MGTOVO and CLUTTR. CLUTIN, INICLT and GETTTD initialize certain radar, clutter and terrain variables. ANGWDT, DETGEO and RANMIN determine the geometry of the clutter areas. SIGMA0 determines the clutter reflectivity coefficient for land and sea terrain. SUMCLT computes the total clutter power for each clutter range and MGTOVO converts clutter power sums to complex voltages. CLUTTR is the orchestration routine and controls the clutter calculations and operations. All subroutines used for this FE are described in Table 2.12-2.

TABLE 2.12-2. Subroutine Descriptions.

MODULE NAME	DESCRIPTION
ANGWDT	Compute azimuth interval between min and max azimuth limits.
CLUTIN	Initializes several clutter coefficients for sea clutter reflectivity.
CLUTTR	Main clutter subroutine, directs clutter calculation.
DETGEO	Determines the boundaries of the clutter patches, both range and azimuth.
GETTTD	Initializes certain clutter terrain variables, calculates triangular terrain data for surface point.
INICLT	Initializes several radar specific clutter parameters.
MGTOVO	Changes clutter power sums to complex voltages.
RANMIN	Calculates minimum possible range for clutter contributions.
SIGMA0	Determines clutter reflectivity of current clutter cell using Georgia Institute of Technology data or Lincoln Lab data. Also computes terrain visibility due to shadowing.
SUMCLT	Determines total clutter power from all clutter cell contributors, one range at a time.

2.12.2 Verification Design Elements

Design elements defined for the Clutter Signal FE are listed in Table 2.12-3. A design element is a feature or an algorithm that represents a specific component of the FE design. The first eight design elements are fully described in section 2.12.2 of the ASP II for ESAMS.

TABLE 2.12-3. Clutter Design Elements.

SUBROUTINE	DESIGN ELEMENT	DESCRIPTION
INICLT SUMCLT	12-1 Clutter Signal in Sum and Difference Channels	Sums clutter power from multiple clutter cells for azimuth, elevation and sum channels.
INICLT DETGEO RANMIN	12-2 Range Ambiguity	Computes at which ambiguous ranges clutter will be returned based on radar PRI.
ANGWDT DETGEO SUMCLT	12-3 Clutter Patch Area	Determines surface area of terrain from which clutter is returned. Defined by range, range width and azimuth limits.
CLUTIN SIGMA0	12-4 Clutter Reflectivity - Native Mode	Determines sea clutter reflectivity using Georgia Tech empirical equations.
SIGMA0 GETTTD	12-5 Land Clutter Coefficient - Native Mode	Determines land clutter reflectivity using Georgia Tech or Lincoln Lab data.
SIGMA0	12-6 Terrain Shadowing - Native Mode	Calculates the percent of terrain in shadow, i.e. no clutter contribution, using a statistical method.
MGTOVO	12-7 Clutter Power Summation and Conversion	Converts 3-channel clutter powers to complex voltages.
SUMCLT	12-8 Coordinate Conversion for GRACE	Converts ESAMS coordinate system into that used by GRACE clutter model.

TABLE 2.12-3. Clutter Design Elements. (Contd.)

SUBROUTINE	DESIGN ELEMENT	DESCRIPTION
INICLT CLUTTR	Initialization of Clutter Variables	Constants involved in clutter calculations are initialized.
CLUTTR MGTOVO	Clutter Management	Directs clutter calculations, i.e. determines when they should be performed based on clutter authorization from user input, timing and other conditions.

2.12.3 Desk Checking Activities and Results

The code implementing this FE was manually examined using the procedures described in Section 1.1 of this report. Any discrepancies discovered are described in the Table 2.12-4.

TABLE 2.12-4. Code Discrepancies.

DESIGN ELEMENT	DESK CHECK RESULT
12-1 Clutter Signal in Sum and Difference Channels	D1. By making the number of intervals an odd number with one centered on the antenna beamwidth, the center interval will have an antenna voltage gain of zero resulting in no clutter return from the center patch.
12-2 Range Ambiguity	D2. An error in determining if minimum possible range is within the horizon (line 94 of RANMIN) will indicate range is within horizon when it is not, i.e. SINEL = 0.0. Equation should be: IF (ZANT(IRADAR) .LT. (-RANHOR(IRADAR) * SINEL)) THEN. D3. A ground range is computed as a minimum possible range and not a slant range at line 100 of RANMIN. Equation should be: RNGMIN = MAX(RANGMN(IRADAR), -ZANT/SINEL) D4. The speed-of-light is missing from an equation which determines if minimum range ambiguity is less than minimum physical range.
12-4 Clutter Reflectivity - Native Mode	D5. The exponent on the wavelength term, WLF, is 0.25 and not 1.25 as required by the references.
12-5 Land Clutter Coefficient - Native Mode	D6. The values in a MSLD*.DAT data file for constant A in Georgia Tech empirical equation do not match references. D7. If the user input variable for default terrain height, TERZ, is utilized, the clutter geometry will be computed using a different antenna height than the clutter antenna response and the terrain visibility factor.

Except as noted in Table 2.12-5, overall code quality and internal documentation were evaluated as good. Subroutine I/O were found to match the ASP II descriptions.

TABLE 2.12-5. Code Quality and Internal Documentation Results.

SUBROUTINE	CODE QUALITY	INTERNAL DOCUMENTATION
CLUTIN	OK	The purpose should be more descriptive by stating that this routine computes constants for sea clutter only. The individual equations should be commented. The constant 0.63 which is multiplied by wave height is not in the referenced material and is not explained in the code.
CLUTTR	OK	Discussion under METHOD says CLUTTR calls INICLT which is not true. INICLT is called by GRAMI. The variables IBIN, NEWCLT and SUMMAG are missing from the Argument Description. TGOLD and IERR should be added to Local Variables. CLUTDB should be deleted from ENVRN. IRADFL should be added to FLAGS. ATNMTI should be added to GRADAR. CLTEPS should be added to MULC.
GETTTD	The array VECNRM does not appear to be used anywhere.	There are no Header entries for this routine, e.g. Abstract, Purpose, etc. The comment on line 24, “find altitude and eventually other things” is amusing but not very specific. (It appears this routine was written to work but was never revisited for completion.)
INICLT	Line 120 appears to be unnecessary code because GDT should never be 0.0. RWIDTH does not appear to be used anywhere else in the code. PWCL(IAREA,J) is recalculated in subroutine CLUTTR.	The variables DELEL, CELMAX and CELMIN are not listed under the MULC common block descriptions. (These variables are used for Quick Clutter which is projected to be removed from the code by version 2.7)
MGTOVO	The code which generates the random aspects of PHASE is commented out without explanation. The actual lines of operable code are few compared to the code commented out.	The METHOD describes how the code would operate if the random aspects of PHASE were operable. There should be a note explaining the current operation of the code which sets PHASE = 0.
SIGMA0	At line 87 the value of ANGI, grazing angle, is computed and prevented from being less than 0. Line 90 is an IF statement for which the ELSE portion is accessed when ANGI is less than 0, which will never happen.	ZDIFF needs to added to Local Variables. The explanation of EXP2 under Subroutines Called does not make sense.
SUMCLT	The equations for summing total clutter power DFAPOW, DFEPOW and SUMPOW (lines 240-245) could reduce RANGE/RANGE ** 4 to 1/RANGE ** 3 if explained with a comment.	The Abstract should indicate that SUMCLT sums clutter for one range ring at a time. PI and PI2 are not listed under Parameters. XTILT and YTILT should be deleted from Local Variables. ALPOFF is not listed under GRADAR. Add: COMMON /TERNDR/ IGRACE GRACE code in use flag Delete subroutine LOOKUP from subroutines called.

2.12.4 Software Test Cases

All software testing was performed by running the entire ESAMS model in debug mode. For these tests, ESAMS was run using a sample input file for the specific missile of interest.

The ESAMS8.INP input file used, unless additional changes are specified, was:

```
SXT       0.          -8000.0     200.0
SVT       200.0       90.
IMTION    1  1  1  1
RNOISE    1
KCFLC     0  0  0
          1  0  1
          1  0  1
          1  0  1
```

TABLE 2.12-6. Software Test Cases for Clutter.

Test Case ID	Test Case Description
12-1	OBJECTIVE: Check manipulation of CLINTV(J,I) in INICLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine INICLT at line 112. 3. Step through 2000 and 1000 Do Loops, independently determining proper execution path for each value of CLINTV(J,I) with actual execution path. VERIFY: ESAMS modifies CLINTV as expected. RESULT: OK
12-2	OBJECTIVE: Check calculation of angle off boresight limits array elements in INICLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine INICLT, step to line 130. 3. Observe values of ANGOFF(1-4) and ANGOFR(1-4) on lines 126-129. 4. Step to line 136, observe the values of COSANG(1-4) on lines 130-133. 5. Independently calculate values of the cosines on the ANGOFF(1-4) angles and compare to ESAMS calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK

DRAFT

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-3	OBJECTIVE: Check calculation of range to horizon and power constant in INICLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine INICLT and step to line 140. 3. Observe the values of HORCON, ZSJ and RANHOR on line 136. 4. Independently calculate a value of HORCON from EQ [2.12-7] and compare to ESAMS value. 5. Independently calculate value of RANHOR using values from step 2 and compare to ESAMS value. 6. Step to line 146 and observe the values of PWRTX(1&2), WVLTX(1&2), XLOSS(1&2) and POWCON(1&2) on lines 140 and 141 respectively. 7. Observe the values of POWCON(3), PWRTX(ILUMR), WVLTX(ILUMR) and XLOSS(3) on line 142. 8. Independently calculate values of POWCON(1-3) using the appropriate portions of EQ [2.12-2] and the values above and compare to ESAMS calculated values. VERIFY: ESAMS values match the independently calculated values. RESULT: Value of REARTH used in HORCON not the same as REARTH referenced in code (CONST.INC). Calculations are correct otherwise. XLOSS is not squared in Equations in VSR or PD^3 but is squared in the coded equations.
12-4	OBJECTIVE: Check calculation of minimum range and pulse width of area clutter in INICLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine INICLT and step to line 154. 3. Observe the values of SPDLGT, TSWTCH(1&2) and RANGMN(1&2) on lines 150 and 151 respectively. 4. Independently calculate values of RANGMN(1&2) using values from step 2 and compare to ESAMS calculated values. 5. Step to line 159 and observe the values of PWCL(IAREA, 1-3) and ensure that they are equivalent to TRGW(1-3). VERIFY: ESAMS values match the independently calculated values. RESULT: OK. The variable TSWTCH is not in the RDRD8.DAT data file resulting in a value for TSWTCH for the tested system of zero. Preliminary developer response is that TSWTCH is not for this system.
12-5	OBJECTIVE: Confirm TLASTC array is filled with a large negative number in INICLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine INICLT and step to line 174. 3. Observe and compare the values of BIGNUM, TLASTC(1,1-3) and TLASTC(3,1-3). VERIFY: TLASTC array is filled with large negative numbers. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-6	OBJECTIVE: Check calculation of wavelength in feet and wave factor variables in CLUTIN. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine CLUTIN and step to line 57. 3. Observe the values of WVLTX(IRADFL), WFL, FTM, WAVEHT and HAV on lines 55 and 56. 4. Compare value of FTM to independent value of feet-to-meters conversion of 0.3048. 5. Independently calculate values of WLF and HAV using values from step 2 and compare to ESAMS values. VERIFY: ESAMS values match the independently calculated values. RESULT: OK. The constant 0.63 is not explained and is not in the references.
12-7	OBJECTIVE: Check calculation of several sea clutter variables in CLUTIN. PROCEDURE: 1. Run ESAMS using ESAMS8.INP. 2. Break in subroutine CLUTIN and step to line 58. 3. Observe the values of WLF, HAV and FAC1. 4. Independently calculate the value of *qw* using EQ [2.12-26], which corresponds to FAC1, and compare to FAC1 from step 2. 5. Step to line 59 and observe the value of C4 on line 58. 6. Independently calculate the value of σ_φ, omitting γ, using EQ [2.12-28], raising it to the fourth power, which corresponds to C4, and compare to C4 from step 4. 7. Step to line 60 and observe the value of C5 on line 59. 8. Independently calculate the value of V_w using EQ [2.12-25], which corresponds to C5, and compare to C5 from step 6. 9. Step to line 61 and observe the value of C5 on line 60. 10. Independently calculate the value of A_w using EQ [2.12-24] and the values of C5 from step 6 in place of V_w and FAC1 from step 2 in place of *qw* and compare to C5 from step 8. 11. Step to line 63 and observe the value of C5 on line 61. 12. Independently calculate the modified value of σ^0; omitting $\gamma^{0.4}$, $(\gamma + 0.001)^{0.29}$ and A_i, using EQ [2.12-23] and the values of C5 from step 8 in place of A_w and compare to C5 from step 10. VERIFY: ESAMS values match the independently calculated values. RESULT: Wavelength, λ, is raised to 0.25 and not 1.25 as referenced.
12-8	OBJECTIVE: Check setting of TGOLD on first clutter call in CLUTTR. PROCEDURE: 1. Modify ESAMS8.INP file to compute clutter for both acquisition and track. Run ESAMS. 2. Break in CLUTTR and step to line 143. 3. Note the values of IAREA, IRADAR, TLASTC(IAREA, IRADAR) and BIGNUM. 4. Independently evaluate the response of the IF statement. 5. Step to and observe the next line of code. 6. Observe the value of TIMEG and TGOLD(IRADAR). 7. Continue running model, break in CLUTTR on next pass. 8. Repeat steps 2 through 5. 9. Continue running model until IRADAR = 2 (signifying switch to track). 10. Repeat steps 2 through 5. VERIFY: 1. TGOLD(IRADAR) is set equal to TIMEG for first pass only. 2. TGOLD(IRADAR) is set equal to TIMEG when IRADAR changes. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-9	OBJECTIVE: Check implementation of KCLFL user input array in CLUTTR. PROCEDURE: 1. Run ESAMS using modified ESAMS8.INP input file from test 12-8. 2. Break in subroutine CLUTTR and step to line 150. 3. Observe the value of KCLFL(IAREA,IRADAR) and compare to input value. 4. Independently evaluate the response of the IF statement and observe execution. 5. Continue running model, repeat step 2. 6. Change the value of IRADAR to 2. 7. Repeat steps 3 and 4. VERIFY: Clutter is activated for both acquisition and track. RESULT: OK.
12-10	OBJECTIVE: Check clutter phase operation in IF statement in CLUTTR. PROCEDURE: 1. Return ESAMS8.INP file to original form, implementing clutter for track only. 2. Run ESAMS using ESAMS8.INP. 3. Break in subroutine CLUTTR and step to line 152. 4. Note values contained within IF statement and independently determine the response. [On first pass through CLUTTR, (in track mode) TGOLD(2) = TIMEG]. 5. Observe the next executable line of code (should be line 156). 6. Pass through CLUTTR five times. 7. Repeat steps 3 and 4. [All three conditions should be false] 8. Observe the next executable line of code (should be 185). 9. Repeat step 3. 10. Change value of TLASTC(1,2) to the value of -BIGNUM. 11. Repeat step 4. 12. Observe the next executable line of code (should be 156). 13. Repeat step 3. 14. Change value of TIMEG to 2.5. 15. Observe the next executable line of code (should be 156). VERIFY: Proper operation in phase of clutter IF statement. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-11	OBJECTIVE: Check clutter phase operation in CLUTTR when clutter is beyond the horizon. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine CLUTTR and step to line 152. 3. Deposit value of 2.0 into TIMEG. 4. Step to line 158 and observe the values of SUMMAG, DFAMAG and DFEMAG being set to 0.0. 5. Step to line 164 and observe the value of IDCLUT(1,2) on line 160. 6. Step to line 166 and observe the value of NOAREA (should be 1, set in DETGEO due to range to clutter). 7. Observe the next executable line of code (should be 175 then 177 which sets NEWCLT = .TRUE.). 8. Observe the next executable line of code (should be 189, ignoring DEBUG code). 9. Step to line 195 and observe the values of DFCLAZ, DFCLEL and SUMCL returned from subroutine MGTOVO (should all be 0.0). 10. Step to line 199 and observe the equivalence of RSCL to RGATES on line 195 and PWCL to TRGW on line 197. VERIFY: 1. Value of IDCLUT(1,2) is 21. 2. NOAREA = 1. 3. Next executable line of code after 166 is 175. 4. NEWCLUT is set to TRUE. 5. Next executable line of code after 177 is 189. 6. Values of DFCLAZ, DFCLEL and SUMCL are 0.0. 7. RSCL(1,2) = RGATES(2) and PWCL(1,2) = TRGW(2). RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-12	OBJECTIVE: Check clutter phase operation in CLUTTR when clutter is within the horizon. PROCEDURE: 1. Modify ESAMS10.INP input file as follows: SXT 6000.0 0.0 100.0 SVT 180.0 180.0 RNOISE 1 KCLFL 0 0 0 1 0 1 0 0 0 0 0 0 2. Break in subroutine CLUTTR at line 156 (first calculation of clutter) 3. Note value of TIMEG on line 152, should be ~2.0. 4. Step to line 166 and observe the value of NOAREA (should be 0, set in DETGEO due to range to clutter). 5. Observe the next executable line of code (should be 169, call to SUMCLT). 6. Step to line 175, note the value of NOAREA. 7. Observe the next executable line of code (should be 162). 8. Repeat steps 4, 5 and 6 until NOAREA = 1. 9. Step to next executable line of code. (should be 177 which sets NEWCLUT = .TRUE.) 10. Step to line 195 and observe the values of DFCLAZ, DFCLEL and SUMCL returned from subroutine MGTOVO. 11. Break in subroutine CLUTTR after five passes, TIMEG should not satisfy IF statement, step to line 195 and observe the values of DFCLAZ, DFCLEL and SUMCL returned from subroutine MGTOVO. VERIFY: 1. NOAREA = 0 for several passes due to several range rings. 2. Next executable line of code after 166 is 169. 3. DETGEO and SUMCLT are called until NOAREA = 1. 4. NEWCLUT is set to TRUE. 5. Values of DFCLAZ, DFCLEL and SUMCL are non-zero. 6. Values of DFCLAZ, DFCLEL and SUMCL remain constant when new clutter is not calculated. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-13	OBJECTIVE: Check calculation of azimuth limits of clutter patch in DETGEO. PROCEDURE: 1. Check value used for speed of light, SPDLGT in CONST.INC. 2. Run ESAMS using modified ESAMS10.INP file from test 12-12. 3. Break in subroutine DETGEO and step to line 111. 4. Deposit the following values in RGATES(2), IAMBIG, PRI(IPRI(2),2): RGATES(2) = 18000.0 IAMBIG = -2 PRI(IPRI(2),2) = 5 E-05 5. Independently calculate the value of RANGE using EQ [2.12-4] and [2.12-5] in ASP II and values from steps 1 and 3. 6. Step to line 115, note the value of RANGE on line 111 and compare to independently calculated value. 7. Observe the value of RANHOR(2), should be greater than RANGE. 8. Step to line 118, note the value of ZSJ. 9. Independently calculate the value of G_r using EQ [2.12-12] in ASP II inserting values of RANGE and ZSJ for *R* and *h* respectively. 10. Step to line 121, note the value of GRANGE on line 118 and compare to independently calculated values. 11. Observe the values of COSANG(2) and BOREEL(2). 12. Independently calculate X_c using EQ [2.12-11] in ASP II inserting the value of RANGE for R, COSANG(2) for γ, and BOREEL(2) for θ. 13. Step to line 125 and observe the value of XC on line 121 and compare to independently calculated value. 14. Independently calculate X_c/G_r (from EQ [2.12-13] in ASP II) using XC from step 12 for X_c and GRANGE from step 9. 15. Step to line 128 and observe the value of COSAZM on line 125 and compare to independently calculated value. 16. Note the value of ABS(COSAZM), should be less than 1.0. 17. Step to line 134, independently calculate the inverse cosine of COSAZM and compare to AZIMUTH on line 131. 18. Note value of BOREAZ(2) on line 134. 19. Independently calculate φ_{min} and φ_{max} using EQ [2.12-14] and [2.12-15] respectively, inserting value of BOREAZ(2) for θ_{az} and AZIMUTH for ß/2. 20. Step to line 138 and observe the value of AZMIN and AZMAX on lines 134 and 135 respectively and compare to φ_{min} and φ_{max} from step 18. 21. Step to subroutine CLUTTR, observe the values of AZMAX, AZMIN, GRANGE and RANGE and compare to appropriate values calculated above. VERIFY: ESAMS values match independently calculated values. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-14	OBJECTIVE: Check setting of NOAREA flag in DETGEO. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine DETGEO and step to line 115. 3. Observe the values of RANGE and RANHOR(2). 4. If RANGE is less than RANHOR(2), change its value to be greater than RANHOR(2). 5. Observe the next executable line of code, should be line 150 which sets NOAREA = 1. 6. Repeat steps 1 and 2. 7. Deposit a value for RANGE which is less than RANHOR(2). 8. Step to line 125, deposit value of -1.1 into COSAZM. 9. Step to line 128 and observe the next executable line of code, should be line 141 which sets NOAREA = 1. VERIFY: 1. If RANGE is greater than RANHOR, NOAREA is set to 0.0. 2. If absolute value of COSAZM is greater than 1.0, NOAREA is set to 0.0. RESULT: OK.
12-15	OBJECTIVE: Check minimum range calculations when minimum range is within horizon yet greater than minimum range due to switching in RANMIN. PROCEDURE: 1. Run ESAMS using modified ESAMS10.INP file. 2. Break in subroutine RANMIN and step to line 87. 3. Deposit the following values in BOREEL(2) and ANGOFF(2): BOREEL(2) = 0.035 rad (~ 2°) ANGOFF(2) = 10 4. Step to line 90, independently calculate ELMIN and compare to ELMIN on line 87. 5. Step to line 94, independently calculate the cosine and sine of ELMIN and compare to COSEL and SINEL on lines 90 and 91. 6. Independently calculate minimum range using EQ [2.12-3] from ASP II inserting the value of ZANT(2) for h_a and SINEL for $\sin\theta_{EL}$. 7. Calculate minimum range obtained using (ZANT(2) * COSEL) and compare to minimum range from step 6. 8. Evaluate response of IF statement using equation as coded and corrected equation. 9. Observe the next executable line of code, should be line 97, setting NOAREA = 0. 10. Step to line 100, note value of RANGMN(2). 11. Independently calculate -ZANT(2) * COSEL/SINEL and compare to R_{min} from step 6. 12. Step to line 103 and note value of RNGMIN. Independently evaluate the value of RNGMIN and compare. VERIFY: 1. Minimum range due to geometry is calculated and compared to horizon range correctly. 2. Minimum range due to switching time is calculated correctly. RESULT: 1. The minimum range due to geometry calculated in the model is smaller than the minimum range calculated using equations from the PD[3]. Equation for minimum range on line 100 should be RNGMIN = MAX (RANGMN (IRADAR), -ZANT (IRADAR)/SINEL). 2. OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-16	OBJECTIVE: Check minimum range calculations when minimum range is greater than horizon range in RANMIN. PROCEDURE: 1. Run ESAMS using modified ESAMS8.INP file. 2. Break in subroutine RANMIN and step to line 87. 3. Deposit the following values in BOREEL(2) and ANGOFF(2): BOREEL(2) = 0.174358 rad (~ 9.9°) ANGOFF(2) = 10 4. Note values of COSEL and SINEL on lines 90 and 91. 5. Independently calculate minimum range using EQ [2.12-3] from ASP II inserting the value of ZANT(2) for h_a and SINEL for $\sin\theta_{EL}$. 6. Compute RANHOR * SINEL. 7. Calculate minimum range obtained using (ZANT(2) * COSEL) and compare to minimum range from step 5. 8. Evaluate response of IF statement using equation as coded and corrected equation. 9. Observe the next executable line of code, should be line 114, setting NOAREA = 1. VERIFY: Minimum range due to geometry is calculated and compared to horizon range correctly. RESULT: Incorrect equation on line 94 allows clutter to proceed when minimum range is beyond horizon.
12-17	OBJECTIVE: Check minimum range calculations when BOREEL approaches ANGOFF in RANMIN. PROCEDURE: 1. Run ESAMS using modified ESAMS8.INP file. 2. Break in subroutine RANMIN and step to line 90. 3. Deposit the following value in ELMIN on line 90: ELMIN = -0.001 4. Independently calculate minimum range using EQ [2.12-3] from ASP II inserting the value of ZANT(2) for h_a and SINEL for $\sin\theta_{EL}$. 5. Step to line 100, note the value of RANGMN. 6. Step to line 103 and note the value of RNGMIN on line 100 and compare to RANGMN from step 5. 7. Repeat step 3 depositing -0.00001 in ELMIN. 8. Repeat steps 4 and 6. VERIFY: Minimum range due to geometry is calculated correctly. RESULT: Model crashes due to floating overflow caused by SINEL in denominator when ELMIN = -0.00001. If the equation on line 94 (test 12-16) were corrected this would be prevented.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-18	OBJECTIVE: Check minimum range calculations when ANGOFF is large in RANMIN. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine RANMIN and step to line 87. 3. Deposit the following values in BOREEL(2) and ANGOFF(2): BOREEL(2) = 0.1 rad (~ 5.7°) ANGOFF(2) = 100. 4. Independently calculate minimum range using EQ [2.12-3] from ASP II inserting the value of ZANT(2) for h_a and SINEL for $\sin\theta_{EL}$. 5. Step to line 100, note the value of RANGMN. 6. Step to line 103 and note the value of RNGMIN on line 100 and compare to RANGMN from step 7. VERIFY: Minimum range due to geometry is calculated correctly. RESULT: OK.
12-19	OBJECTIVE: Check minimum range calculations when minimum range is within horizon and less than minimum range due to switching in RANMIN. PROCEDURE: 1. Run ESAMS using modified ESAMS8.INP file. 2. Break in subroutine RANMIN and step to line 87. 3. Deposit the following values in BOREEL(2) and ANGOFF(2): BOREEL(2) = -0.035 rad (~ 2°) ANGOFF(2) = 50.0 4. Step to line 90, independently calculate ELMIN and compare to ELMIN on line 87. 5. Step to line 94, independently calculate the cosine and sine of ELMIN and compare to COSEL and SINEL on lines 90 and 91. 6. Independently calculate minimum range using EQ [2.12-3] from ASP II inserting the value of ZANT(2) for h_a and SINEL for $\sin\theta_{EL}$. 7. Deposit the value of 25 in RANGMN. 8. Step to line 103 and note the value of RNGMIN on line 100 and compare to RANGMN from step 7. VERIFY: 1. Minimum range due to geometry is calculated and compared to horizon range correctly. 2. Minimum range due to switching time is calculated correctly. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-20	OBJECTIVE: Check calculation of ambiguity index if minimum range ambiguity is within minimum range in RANMIN. PROCEDURE: 1. Run ESAMS using modified ESAMS8.INP file. 2. Break in subroutine RANMIN and step to line 103. 3. Deposit the following values: RNGMIN = 100. RGATES(2) = 7000.0 PRI = 2 E-05 4. Independently calculate the value of IAMBIG using values from step 3. 5. Step to line 106 and note the value of IAMBIG on line 103 and compare to independent calculation. 6. Independently calculate R_i using EQ [2.12-5], inserting the value of RGATES(2) in for R_g, IAMBIG in for i and (SDPLGT * PRI) /2. in for R_u. 7. Step to line 114 and compute (RGATES(2) + IAMBIG * PRI(IPRI(2),2)/2.) and compare to R_i from step 6. 8. Note the value of RNGMIN on line 108 and compare to the value from step 7. 9. Independently evaluate response of IF statement and compare to operation in code. VERIFY: 1. Ambiguity index is correctly computed. 2. Comparison of minimum ambiguous range to minimum range is correct. RESULT: Proper check not made because speed-of-light is missing from equation implementation in code.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-21	OBJECTIVE: Check clutter calculation when minimum ambiguous range is close to minimum physical range in RANMIN. PROCEDURE: 1. Run ESAMS using modified ESAMS8.INP file. 2. Break in subroutine INICLT and deposit the following values: ANGOFF(1-4) = 15. lines 126-129 ZSJ = 10.0 line 136 3. Break in subroutine DETGEO and step to line 99. 4. Break in subroutine RANMIN and step to line 87. 5. Deposit the following value: BOREEL(2) = 0.15708 6. Step to line 100 and deposit the following value: ZANT = 10. 7. Step to line 103 and deposit the following values: RGATES(2) = 3842.0 PRI(IPRI(2),2) = 5×10^{-6} 8. Step to line 107 and note value of IAMBIG on line 103, should be -4. 9. Return to subroutine DETGEO and step to line 115 and note the value of RANGE, should be 844.076. 10. Step to line 128 and note the value of COSAZM, should be 0.979911. 11. Repeat steps 2 through 5. 12. Step to line 103 and deposit the following values: RGATES(2) = 3843.0 PRI(IPRI(2),2) = 5×10^{-6} 13. Step to line 107 and note value of IAMBIG on line 103, should be -5. 14. Return to subroutine DETGEO and step to line 115 and note the value of RANGE, should be 95.5945. 15. Step to line 128 and note the value of COSAZM, should be 1.00002. 16. Repeat steps 2 through 5. 17. Step to line 103 and deposit the following values: RGATES(2) = 3844.0 PRI(IPRI(2),2) = 5×10^{-6} 18. Step to line 107 and note value of IAMBIG on line 103, should be -5. 19. Return to subroutine DETGEO and step to line 115 and note the value of RANGE, should be 96.5945. 20. Step to line 128 and note the value of COSAZM, should be 0.9997346. VERIFY: 1. Clutter is calculated at ranges 3842 and 3844. 2. Clutter not calculated at range 3843 (error discovered in desk-check). RESULT: Clutter is not calculated at range 3843 due to error in calculating minimum possible range confirming error found in desk checking.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-22	OBJECTIVE: Check updating of TLAST in MGTOVO. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine MGTOVO and step to line 84. 3. Observe the values of TIMEG, TLAST and TAUCLT. 4. Independently evaluate the response of the IF statement and determine the next executable line of code. 5. Step to next executable line of code and compare to independent evaluation. 6. Repeat step 2. 7. Deposit a value into TIMEG which would have the opposite effect on the response of the IF statement. 8. Repeat step 5. VERIFY: Value of TLAST is set equivalent to TIMEG when time since last call to MGTOVO is greater than correlation time. RESULT: OK. MGTOVO is also called from subroutine ACQEXE even when clutter is not being calculated, i.e., during acquisition. Dummy numbers are used as clutter variables in the MGTOVO routine.
12-23	OBJECTIVE: Check calculation of complex voltage sums for all 3-channels in MGTOVO. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine MGTOVO, after call to SUMCLT, and note the values of PCLTAZ, PCLTEL and PCLTSM. 3. Step to line 104 and note the value of PHASE, should be 0.0. 4. Independently evaluate the values of DAZCLT, DELCLT and SMCLT and compare to values on lines 107, 108 and 109 respectively. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-24	OBJECTIVE: Check calculation of angular difference for 0 to 2B case in ANGWDT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine ANGWDT and step to line 46. 3. Deposit the following values: ANGLE1 = 1.745 ANGLE2 = 1.222 4. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46. 5. Step to next executable line of code, should be line 63. 6. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value. VERIFY: ESAMS values match independently calculated values. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-25	OBJECTIVE: Check calculation of angular difference for arc across azimuth = 0^0 in ANGWDT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine ANGWDT and step to line 46. 3. Deposit the following values: ANGLE1 = 0.35 ANGLE2 = -0.175 4. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46. 5. Step to next executable line of code, should be line 63. 6. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value. 7. Repeat step 2. 8. Deposit the following values: ANGLE1 = 6.46 ANGLE2 = 5.76 9. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46. 10. Step to next executable line of code, should be line 63. 11. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-26	OBJECTIVE: Check calculation of angular difference for ARG2B ARG1 in ANGWDT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break in subroutine ANGWDT and step to line 46. 3. Deposit the following values: ANGLE1 = 0.35 ANGLE2 = 0.50 4. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46. 5. Step to next executable line of code, should be line 52. 6. Step to next executable line of code, should be line 54 because ANGWDT < 0. 7. Independently calculate value of ANGWDT for less than 0 case. 8. Step to line 60, note the value of ANGWDT on line 54. 9. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value. VERIFY: ESAMS values match independently calculated values. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

<table>
<tr><th>Test Case ID</th><th>Test Case Description</th></tr>
<tr><td>12-27</td><td>OBJECTIVE: Check calculation of angular difference greater than 2B in ANGWDT.
PROCEDURE:
1. Run ESAMS using ESAMS8.INP file.
2. Break in subroutine ANGWDT and step to line 46.
3. Deposit the following values:
ANGLE1 = 4.0
ANGLE2 = -4.0
4. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46.
5. Step to next executable line of code, should be line 58 because ANGWDT > 2B.
6. Independently calculate value of ANGWDT for greater than 2B case.
7. Step to line 60, note the value of ANGWDT on line 58.
8. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value.
VERIFY: ESAMS values match independently calculated values.
RESULT: OK.</td></tr>
<tr><td>12-28</td><td>OBJECTIVE: Check calculation of angular difference for angle close to 2B case in ANGWDT.
PROCEDURE:
1. Run ESAMS using ESAMS8.INP file.
2. Break in subroutine ANGWDT and step to line 46.
3. Deposit the following values:
ANGLE1 = 6.283185307
ANGLE2 = 0.0000009
4. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46.
5. Step to next executable line of code, should be line 63.
6. Step to line 66 and independently evaluate the response of the IF statement.
7. Step to next executable line of code, should be line 68, and note value of EPSILN.
8. Independently calculate new value of ANGWDT and compare to value on line 68.
9. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value.
10. Repeat step 2.
11. Deposit the following values:
ANGLE1 = 6.283185307
ANGLE2 = -0.0000009
12. Step to line 50, independently calculate the value of ANGWDT and compare to value on line 46.
13. Step to next executable line of code, should be line 58.
14. Independently calculate new value of ANGWDT and compare to value on line 58.
15. Return to subroutine SUMCLT and note the value of ANGWDT and compare to independently calculated value.
VERIFY: ESAMS values match independently calculated values.
RESULT: OK.</td></tr>
</table>

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-29	OBJECTIVE: Check grazing angle calculation in SIGMA0. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine SIGMA0 and step to line 87. 3. Deposit the following values: ZANT = 10. ZCLUT = 40.0 RCLT = 2000.0 4. Knowing that EPSILN = 1 x 10^{-6}, independently calculate the value of ANGI. 5. Step to line 90 and compare value of ANGI to independently calculated value. 6. Repeat step 2. 7. Deposit the following values: ZANT = 10.0 ZCLUT = 4.0 RCLT = 2000.0 8. Repeat steps 4 and 5. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-30	OBJECTIVE: Check calculation of sea clutter reflectivity in SIGMA0. PROCEDURE: 1. Add ITTYPE = 1 (sea) to ESAMS8.INP file. 2. Run ESAMS using ESAMS8.INP file. 3. Break into subroutine SIGMA0 and step to line 90 and set ANGI equal to 0.02. 4. Step to line 93, note the value of C4. 5. Step to line 94, independently calculate the value of FAC1 on line 93 and compare to independent calculation. 6. Step to line 95, independently calculate the value of FAC1 on line 94 and compare to independent calculation. 7. Note value of C5. 8. Step to next line, independently calculate the value of REFLEC on line 95 and compare to independent calculation. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-31	OBJECTIVE: Check values of constants A, B, C and D from Georgia Tech empirical equation in SIGMA0. PROCEDURE: 1. Independently calculate the values of A, B, C and D for all terrain types using the frequency of the tracker for the system of interest. 2. Run ESAMS using ESAMS8.INP file. 3. Break into subroutine SIGMA0 and step to line 98 and deposit the value of 2 into ITTYPE. 4. Step to line 100, note the values of A, B, C and D and compare to independent calculations. 5. Repeat steps 3 and 4, substituting the values of 3, 4, 5, 6, 7, 8, and 9 into ITTYPE. VERIFY: ESAMS values match independently calculated values. RESULT: Values retrieved for A do not agree with independent calculations based on source documentation.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-32	OBJECTIVE: Check calculation of land clutter reflectivity using the Georgia Tech equation in SIGMA0. PROCEDURE: 1. Add ITTYPE = 2 and WET = 1 to ESAMS8.INP file. 2. Run ESAMS using modified ESAMS8.INP file. 3. Break into subroutine SIGMA0 and step to line 100 and note the values of A(2), ANGI, C(2), B(2), D(2) and RMSTRN. 4. Independently calculate the value of F^0 using values from step 3 applied to EQ [2.12-29] from ASP II. 5. Step to line 103, note the value of REFLEC on line 100 and compare to F^0. 6. Independently calculate wet terrain reflectivity for terrain type two using F^0. 7. Step to line 107, note the value of REFLEC on line 105 and compare to wet terrain F^0. 8. Repeat step 2. 9. Set ITTYPE = 3. 10. Repeat steps 3, 4, 5, 6 and 7. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-33	OBJECTIVE: Check calculation of Rural/Low Relief Lincoln Laboratory ground clutter parameters in SIGMA0. PROCEDURE: 1. Add ITTYPE = 10 to ESAMS8.INP file. 2. Run ESAMS using modified ESAMS8.INP file. 3. Break into subroutine SIGMA0 and step to line 111. 4. Observe the next two executable lines of code, should be 112 then 114. 5. Deposit the value of 0.01 into ANGI. 6. Independently calculate the value of REFLEC at ANGI = 0.01 radians. 7. Step to line 124, observe the value of REFLEC on the right side of the equation and compare to the independent calculation. 8. Independently convert REFLEC to absolute, step to next line of code and compare independent calculation with value of REFLEC on line 124. 9. Repeat step 2 and 3. 10. Deposit the value of 0.09 into ANGI. 11. Repeat steps 6, 7 and 8. VERIFY: 1. ESAMS values match independently calculated values from the Rural/Low Relief data tables. 2. Reflectivity in dB is correctly converted to absolute units. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-34	OBJECTIVE: Check calculation of Rural/High Relief Lincoln Laboratory ground clutter parameters in SIGMA0. PROCEDURE: 1. Add ITTYPE = 11 to ESAMS8.INP file. 2. Run ESAMS using modified ESAMS8.INP file. 3. Break into subroutine SIGMA0 and step to line 111. 4. Observe the next two executable lines of code, should be 115 then 117. 5. Deposit the value of 0.06 into ANGI. 6. Independently calculate the value of REFLEC at ANGI = 0.06 radians. 7. Step to line 124, observe the value of REFLEC on the right side of the equation and compare to the independent calculation. 8. Repeat step 2 and 3. 9. Deposit the value of 0.14 into ANGI. 10. Repeat steps 6 and 7. VERIFY: ESAMS values match independently calculated values from the Rural/High Relief data tables. RESULT: OK.
12-35	OBJECTIVE: Check calculation of Urban Type Lincoln Laboratory ground clutter parameters in SIGMA0. PROCEDURE: 1. Add ITTYPE = 12 to ESAMS8.INP file. 2. Run ESAMS using modified ESAMS8.INP file. 3. Break into subroutine SIGMA0 and step to line 111. 4. Observe the next two executable lines of code, should be 118 then 120. 5. Deposit the value of 0.01 into ANGI. 6. Independently calculate the value of REFLEC at ANGI = 0.01 radians. 7. Step to line 124, observe the value of REFLEC on the right side of the equation and compare to the independent calculation. 8. Repeat step 2 and 3. 9. Deposit the value of 0.11 into ANGI. 10. Repeat steps 6 and 7. VERIFY: ESAMS values match independently calculated values from the Urban data tables. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-36	OBJECTIVE: Check calculation of terrain visibility factor in SIGMA0. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine SIGMA0 and step to line 132. 3. Deposit value of 7.5 into ZDIFF. 4. Independently calculate the value of the terrain visibility factor, VIS, from Figure I-5 in the VSR for a range of 10 km and effective radar height of 0-15 m. 5. Step to line 134 and deposit 10000. into RCLT. 6. Step to line 135, note the value of REFLEC (right side) and VIS and compare to VIS from step 4. 7. Step to next line, independently calculate value of F^0 using EQ [2.12-31] inserting value of REFLEC in for F and VIS in for f_{vis} and compare to REFLEC from left side of line 135. 8. Repeat step 2. 9. Deposit value of 45. into ZDIFF. 10. Independently calculate the value of terrain visibility factor, VIS, from Figure I-5 in the VSR for a range of 20 km and effective radar height of 30-60 m. 11. Repeat steps 5, 6 and 7. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-37	OBJECTIVE: Check model operation with out-of-bounds terrain type. PROCEDURE: 1. Add illegal terrain type, ITTYPE = 14, to ESAMS8.INP file. 2. Run ESAMS in non-debug mode and observe operation. VERIFY: ESAMS operation suspended due to illegal terrain type. RESULT: Model crashes due to a divide by zero.
12-38	OBJECTIVE: Check initialization of terrain characteristics in GETTTD. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine GETTTD and step to line 14. 3. Note the value of ZPT on line 11 and compare to default value of 0 (from TERZ in PROGC). 4. Note the value of ITRTYP on line 12 and compare to default value of 3 (from TTYPE in PROGC). 5. Note the value of RUFNES on line 13 and compare to default value of 1.0 (from TERRF in PROGC). 6. Cancel model operation, modify ESAMS8.INP with the following values: TERZ = 26.0 TTYPE = 8. TERRF = 6.38 7. Repeat steps 1 and 2. 8. Note the value of ZPT on line 11 and compare to value of TERZ from step 6. 9. Note the value of ITRTYP on line 12 and compare to value of TTYPE from step 6. 10. Note the value of RUFNES on line 13 and compare to value of TERRZ from step 6. VERIFY: 1. Default values for terrain type, terrain height and terrain roughness are correct. 2. User inputs for terrain type, terrain height and terrain roughness are correctly initialized. RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	**Test Case Description**
12-39	OBJECTIVE: Check calculation of clutter intervals within a range ring in SUMCLT. PROCEDURE: 1. Modify ESAMS8.INP as follows: MULC RELEN 1000. 1000. 1000. 1000. END 2. Run ESAMS using modified ESAMS8.INP file. 3. Break into subroutine SUMCLT and step to line 162. 4. Deposit the following values: AZMIN = 1.745 AZMAX = 2.27 5. Step to line 167, deposit the value of 9550 into RANGE and note the values of RELEN(2) and ANGWID. 6. Independently calculate the value of NINTVL. 7. Step to line 170, note the value of NINTVL on line 167 and compare to independently calculated value. 8. Step to next executable line of code, should be line 173 because NINTVL is odd. 9. Repeat steps 3 and 4. 10. Step to line 167, deposit value of 11430 into RANGE. 11. Repeat steps 6 and 7. 12. Step to next executable line of code, should be line 171 because NINTVL is even. 13. Repeat steps 3 and 4. 14. Step to line 167, deposit the value of 3900 into RANGE. 15. Repeat steps 6 and 7. VERIFY: ESAMS values match independently calculated values. RESULT: OK.
12-40	OBJECTIVE: Check geometry of intervals within a clutter range ring in SUMCLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine SUMCLT, step to line 162 and note the values of AZMIN and AZMAX. 3. Step to line 170 and note the values of RANGE and NINTVL. 4. Step to line 178, independently calculate the value of)2$_{AZ}$ using EQ [2.12-19] in ASP II inserting ANGWID in for ß and NINTVL in for N and compare to DAZMUT on line 176. 5. Independently calculate the values of AZIMUTH, XCLUT and YCLUT for INTRVL = 1 and 15. 6. Step through DO 3000 loop and note values of AZIMUTH, XCLUT and YCLUT at appropriate values of NINTVL and compare to independently calculated values. VERIFY: ESAMS values match independently calculated values RESULT: OK.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-41	OBJECTIVE: Check calculation of reflectivity times antenna channel gains in SUMCLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine SUMCLT and step to line 213. 3. Deposit the following values: REFLEC = 0.5 GSUM = 2.778×10^{-3} GAZM = -6.039×10^{-5} 4. Independently calculate the value of DAPW. 5. Step to line 214 and deposit 8.901×10^{-5} into GELE. 6. Independently calculate the value of DEPW and SMPW. 7. Step to line 218 and note the values of DAPW on line 213, DEPW on line 214 and SMPW on line 215. 8. Step to line 223 and note the values of DAPOW, DEPOW and SMPOW on lines 218, 219 and 220. 9. Step to line 235 and execute DO 3000 loop a second time. 10. Repeat steps 3 and 5. 11. Step to line 223 and note the values of DAPOW, DEPOW and SMPOW. VERIFY: 1. ESAMS calculation of DAPW, DEPW and SMPW match independent calculations. 2. Values of DAPOW, DEPOW and SMPOW equal DAPW, DEPW and SMPW respectively on first pass. 3. Values of DAPOW, DEPOW and SMPOW are incremented by DAPW, DEPW and SMPW respectively on successive passes. RESULT: OK.
12-42	OBJECTIVE: Check calculation of reflectivity times antenna channel gains for entire range ring in SUMCLT. PROCEDURE: 1. Modify ESAMS8.INP as follows: (This reduces number of clutter intervals to manageable number to simplify calculations.) MULC RELEN 1000. 1000. 1000. 1000. END 2. Run ESAMS using modified ESAMS8.INP file. 3. Break into subroutine SUMCLT and step to line 170 and note the value of NINTVL on line 167. 4. Step to line 218 and note the values of DAPW on line 213, DEPW on line 214 and SMPW on line 215. 5. Step to line 223 and note the values of DAPOW, DEPOW and SMPOW on lines 218, 219 and 220. 6. Step to line 235 and execute DO 3000 loop a second time. 7. Repeat steps 4 and 5. 8. Step to line 235 and execute DO 3000 loop a third time. 9. Repeat steps 4 and 5. 10. Independently calculate the final values of DAPOW, DEPOW and SMPOW and compare to step 9. 11. Step to line 235 and verify Do loop has expired. VERIFY: All intervals within the clutter patch are accounted for and summed properly. RESULT: The azimuth channel clutter power is zero for the center interval, otherwise, sums handled correctly.

TABLE 2.12-6. Software Test Cases for Clutter. (Contd.)

Test Case ID	Test Case Description
12-43	OBJECTIVE: Check calculation of total clutter power signal for each channel in SUMCLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine SUMCLT and step to line 238. 3. Note the values of TRGW(2) and BOREEL(2). 4. Independently calculate range gate width, R_{WD}, using EQ [2.12-20] inserting the value of TRGW(2) in for J and BOREEL(2) in for 2_{el}. 5. Step to line 240 and note the values of DFPOW, RANGE, DAZMUT, RNGWID, DAPOW and POWCON(2). 6. Compare value of RNGWID with independently calculated value from step 4. 7. Step to line 242, independently calculate the value of DFAPOW and compare to value of DFAPOW from step 5. 8. Note the values DFEPOW and DEPOW and independently calculate the value of DFEPOW. 9. Step to line 244, note the value of DFEPOW on line 242 and compare to independently calculated value. 10. Note the values of SUMPOW and SMPOW and independently calculate the value of SUMPOW. 11. Step to next line, note the value of SUMPOW on line 244 and compare to independently calculated values. VERIFY: ESAMS values match independently calculated values RESULT: OK.
12-44	OBJECTIVE: Check coordinate conversion for GRACE in SUMCLT. PROCEDURE: 1. Run ESAMS using ESAMS8.INP file. 2. Break into subroutine SUMCLT, step to line 188 and deposit the value of 1 into IGRACE. 3. Step to line 196 and deposit the value of 1.396 into AZIMUTH. 4. Step to line 198, independently calculate the value of 2_{aznrth} using EQ [2.12-33] in ASP II inserting AZIMUTH in for 2_{az} and compare to AZNORTH on line 196. 5. Repeat steps 2, 3 and 4, depositing 2.62, 4.695 and 6.02 in for AZIMUTH respectively on each pass through step 3. VERIFY: ESAMS values match independently calculated values RESULT: OK.

2.12.5 Conclusions and Recommendations

2.12.5.1 Code Discrepancies

The calculation of clutter is very complex and requires a large amount of code. It is not surprising then that several errors were uncovered. A detailed description of each follows.

The radar wavelength term, λ, in the Georgia Tech Sea Clutter reflectivity equations is raised to 1.25 (Eq. 2.12-23 in ASP-II) but is incorrectly raised to 0.25 in the code. Also, in computing the average wave height, the significant waveheight is multiplied by 0.63. This constant is not apparent in the referenced documentation and is unexplained in the code. The impact of the first problem is to increase the sea clutter reflectivity. The impact of the second problem, if it is incorrect, is unclear. It is recommended that the first problem be corrected in the code and the second problem be examined to determine if the given methodology is correct in calculating average wave height.

The constant A used in the Georgia Tech Land Clutter reflectivity model is different than what was expected based on the original source reference. The Georgia Tech reference uses the equation 8 log(λ/3.2) (λ is the wavelength) to determine the values of A based on the 9.5 GHz data. In examining the data file which contains the values of A for the system of interest, MULC8.DAT, it was found that the values used by the model do not agree with source reference. Finally, in independently recreating the tabulated data presented in the CMS, the values of A computed for terrain types 3 and 4 at 3 GHz did not agree with the presented data. The impact of this is to generate clutter reflectivity values which do not agree with the accepted model. It is recommended that the data values for A be made to agree with the accepted Georgia Tech land clutter model.

In the equation which checks to determine if the minimum ambiguous range is less than the minimum physical range (due to geometry or switching time) the term for speed-of-light, c, is missing. The impact of this is that clutter originating within minimum physical range will be included in total clutter computations when it should not have been. It is recommended that the speed-of-light term be added to the applicable equation.

The value of the term used to compute the range to the horizon, HORCON, has not been calculated using the 4/3 earth curvature model using the given value of earth radius as specified in the VSR. This will result in the horizon being at a greater distance than it should be, therefore more clutter will be calculated. It is recommended that the horizon constant, HORCON, be changed to the correct value.

In computing minimum range to clutter, the code contains an equation within an IF statement which can be satisfied when the radar antenna is elevated too high to intersect the ground. This error leads to an overflow condition in computing the minimum slant range when the radar elevation angle equals the given angle off boresight of the radar transmit cone. In addition, the slant range is computed incorrectly which results in clutter calculations being inhibited when clutter is present. It is recommended that these equations be corrected.

A clutter patch is divided into an odd number of intervals and the antenna azimuth response is symmetrical about the center; therefore, the clutter signal strength is zero for the center azimuth channel interval. This results in the clutter signal being less than it should be. We recommend that an even number of intervals be considered in dividing up a clutter patch.

If the user input variable for default terrain height, TERZ, is utilized, the clutter geometry will be computed using a different antenna height than the clutter antenna response and the terrain visibility factor. For example, if the antenna height was 20 and TERZ was 15, the clutter geometry would be calculated with the antenna 35 meters above the surrounding terrain while the clutter antenna response would be computed as if the antenna was 5 meters above the surrounding terrain. The impact of this is uncertain but at a minimum it is inconsistent. It is recommended that the usage and application of the system altitude modifier, TERZ, be re-evaluated.

2.12.5.2 Code Quality and Internal Documentation

Except for the discrepancies listed above, code quality is generally good. There were some code quality issues which should be mentioned, however. Subroutine MGTOVO is called by subroutine ACQEXE when clutter is not being calculated. This results in a value for SMCLT which is based on the simulation time when the simulation time has incremented by the correlation time. This does not appear to cause a problem but the purpose is unclear. Another code quality issues deals with clutter management and the utilization of an important geometric relationship. Presently, the clutter routines are entered when the clutter flag has been set (via KCFLC) and the timing is in “phase” with pre-set clutter timing variables. The geometric relationship which should also be used to allow access to the clutter routines is the relationship between the antenna elevation angle (BOREEL) and the beamwidth angle (ANGOFF). If the antenna is pointed too high, (BOREEL _ ANGOFF) clutter is not possible and there is no need to enter the clutter routines. Currently, the clutter routines are accessed when this condition is satisfied and the determination that clutter is not possible is made via other means within the clutter routines. This expends run time unnecessarily when clutter is not possible.

Some of the internal documentation was incomplete or incorrect. Additional comments and corrections are recommended.

2.12.5.3 External Documentation

The *ESAMS 2.5 Analyst’s Manual* [4] section on clutter could be expanded to include a discussion of ESAMS methodology in determining geometry, antenna response and overall clutter management. An explanation of how often clutter is calculated would be helpful. In addition, the Manual should include a discussion of how the user specified altitude modifiers, TERZ and SITES(X,Y,Z) will affect clutter calculations.

The *GRACE User’s Manual* [8] was also referred to but it contains a general discussion of radar clutter and is not very specific in relation to ESAMS clutter implementation.

The User’s Manual discussion of clutter variables should be expanded to include the impact of a particular selection as well as suggested and prescribed limits on user input variables.